

# Perbandingan Antara Algoritma Penghapusan Bising Adaptif LMS dan Adaptif RLS dalam Penghapusan Bising Kendaraan

Sri Arttini Dwi Prasetyowati<sup>1)</sup>, Adhi Susanto<sup>2)</sup>, Thomas Sriwidodo<sup>2)</sup>, Jazi Eko Istiyanto<sup>3)</sup>

<sup>1)</sup>Mahasiswa S3 Jurusan Teknik Elektro, FT UGM, <sup>2)</sup>Staf Pengajar Jurusan Teknik Elektro, FT, UGM, <sup>3)</sup>Staf Pengajar Jurusan Elektronika dan Instrumentasi, MIPA, UGM

## Abstrak

Permasalahan dalam penelitian ini dikhususkan pada perbandingan hasil LMS Adaptif dan RLS Adaptif pada proses peredaman bising dengan sumber bising suara kendaraan bermotor. Perbandingan didasarkan pada besar  $\mu$  (ukuran langkah), luasan perjalanan bobotnya, dan hasil  $\varepsilon$  (galat) yang diharapkan telah terhapus bising kendaraannya dan yang tersisa adalah informasi yang diharapkan.

**Kata Kunci:** LMS adaptif, RLS Adaptif, bobot, galat, ukuran langkah.

## 1. Pendahuluan

Algoritma LMS (*Least Mean Square*) Adaptif adalah algoritma yang paling sederhana ditinjau dari tidak perlu diketahuinya nilai gradien sesaat yang sama dengan arah penurunan MSE dan korelasi antara input dan targetnya. Dalam penelitian ini ditinjau Algoritma Adaptif yang lain, yaitu RLS (*Recursive Least Square*) Adaptif, selanjutnya dibandingkan dengan Algoritma LMS Adaptif. Perbandingan ini untuk mengetahui Algoritma terbaik yang dapat dipakai dalam penghapusan bising dengan sumber bising suara kendaraan bermotor.

Permasalahan pada penelitian ini adalah membandingkan hasil perjalanan bobot, nilai  $\varepsilon$  dan hasil proses Adaptif LMS dengan RLS untuk peredamaan bising kendaraan bermotor.

## 2. Fundamental

Penghapusan bising dengan Algoritma LMS Adaptif memerlukan beberapa parameter dalam proses perhitungannya. Parameter-parameter tersebut tergantung pada jenis bising yang akan diredam, masukan yang dianggap sebagai  $d_k$  (*desired input*), serta  $x_k$  yang menjadi referensnya. Bentuk sederhana yang jabaran analitika seterusnya mudah dilakukan adalah penggabung linear (*Linear Combiner*) berbentuk Tapis Transversal.

Pada dasarnya, proses adaptasi adalah proses penelusuran menuju titik terendah kurve *Mean Square Error* (MSE) yang berbentuk paraboloid yang dapat diterangkan secara ringkas sebagai berikut:

$$\begin{aligned}\varepsilon_k &= d_k - y_k \\ &= d_k - \mathbf{X}_k^T \mathbf{W} \\ &= d_k - \mathbf{W}^T \mathbf{X}_k \\ \varepsilon_k^2 &= d_k^2 + \mathbf{W}^T \mathbf{X}_k \mathbf{X}_k^T \mathbf{W} - 2d_k \mathbf{X}_k^T \mathbf{W}\end{aligned}$$

dengan  $\varepsilon_k$  = galat hasil adaptasi saat iterasi ke-k

$d_k$  = hasil yang diinginkan (*desired output*) saat iterasi ke-k

$y_k$  = keluaran (*output*) pada saat iterasi ke-k

$x_k$  = masukan (*input*) pada saat iterasi ke-k

$w_k$  = bobot pada iterasi ke-k  
MSE dapat dicari sebagai berikut:

$$\begin{aligned} \text{MSE} &= \\ \xi &= E[\varepsilon_k^2] = \\ &= E[d_k^2] + E[\mathbf{W}^T \mathbf{X}_k \mathbf{X}_k^T \mathbf{W}] - 2E[d_k \mathbf{X}_k^T \mathbf{W}] \end{aligned}$$

Karena  $\mathbf{R} = E[\mathbf{X}_k \mathbf{X}_k^T]$  dan  $\mathbf{P} = E[d_k \mathbf{X}_k]$ , maka

$$\begin{aligned} \xi &= E[\varepsilon_k^2] = \\ &= E[d_k^2] + \mathbf{W}^T \mathbf{R} \mathbf{W} - 2\mathbf{P}^T \mathbf{W} \end{aligned} \quad (2.1)$$

dengan  $\mathbf{R}$  adalah auto-korelasi antar masukan  $x_k$ , sedangkan  $\mathbf{P}$  adalah kros-korelasi antara  $d_k$  dan  $x_k$ , sehingga gradien MSE dapat diuraikan menjadi:

$$\nabla_{\mathbf{W}} \xi = \frac{\partial \xi}{\partial \mathbf{W}} \Rightarrow \begin{bmatrix} \nabla_0 \xi \\ \nabla_1 \xi \\ \vdots \\ \nabla_L \xi \end{bmatrix} = \begin{bmatrix} \partial \xi / \partial w_0 \\ \partial \xi / \partial w_1 \\ \vdots \\ \partial \xi / \partial w_L \end{bmatrix}$$

atau

$$\nabla \xi = 2\mathbf{R}\mathbf{W} - 2\mathbf{P} \quad (2.2)$$

Untuk memperoleh nilai MSE terkecil (minimum MSE), maka vektor bobot  $\mathbf{W}$  haruslah suatu nilai yang optimal  $\mathbf{W}^*$ , ketika gradien MSE = 0:

$$\begin{aligned} \nabla \xi &= 0 = 2\mathbf{R}\mathbf{W}^* - 2\mathbf{P} \\ \Rightarrow \mathbf{W}^* &= \mathbf{R}^{-1}\mathbf{P} \end{aligned} \quad (2.3)$$

Berdasar rangkaian penggabung linear, dengan memperhatikan gambaran bentuk paraboloid, makin jauh dari titik terendah, makin curam atau makin besar nilai gradiennya. Ini berarti langkah adaptasi lebih besar bila masih jauh dari titik terendah yang merupakan titik tujuan. Dengan demikian dapat ditentukan langkah adaptasi, yang berwujud perubahan nilai-nilai bobot, yang sebanding dengan nilai gradien sesaat, pada arah penurunan  $\xi$ , yang berarti kebalikan arah gradien atau  $-(\nabla \xi)$ .

Kemudian karena posisi  $\mathbf{W}^*$  sebanding dengan  $\mathbf{R}^{-1}$ , yaitu  $\mathbf{W}^* = \mathbf{R}^{-1}\mathbf{P}$ , pada persamaan (2.3), maka arah adaptasi disesuaikan dengan rumusan ini, maka dapat ditulis suatu bentuk algoritma adaptasi yang sederhana:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \mathbf{R}^{-1} \nabla_{\xi_k} \quad (2.4)$$

dengan  $\mu$  sebagai parameter variasi ukuran langkah. Parameter  $\mu$  ini diperkirakan secara analitis bila matriks kovarians  $\mathbf{R}$  sudah tertentu atau dicari secara coba-coba bila  $\mathbf{R}$  belum pasti wujudnya. Ide dasar penyesuaian dengan  $\mathbf{R}$  adalah penyesuaian dengan posisi sesaat  $\mathbf{W}_k$  pada permukaan unjukkerja.

Berdasar makna geometris nilai-eigen  $\lambda$ ,  $\lambda$  besar berarti pada arah tebing curam. Dapat dipahami bila ukuran langkah yang terkait dengan nilai parameter  $\mu$ , adalah berbanding terbalik dengan  $\lambda$ . Karena nilai  $\mu$  harus positif (langkah maju menuju titik terendah) maka telah diketahui batasan lengkapnya:  $0 < \mu < \frac{1}{\lambda}$ . Maka algoritma berdasar persamaan (2.4) merupakan algoritma terbaik, yang disebut *Algoritma Newton*.

Dalam kasus isyarat  $x_k$  dan  $d_k$  belum diketahui atau belum ada datanya, maka jelas matriks  $\mathbf{R}$  tidak mungkin dipakai dalam algoritma. Karena itu ditulis algoritma lain yang hanya mengandalkan nilai gradien:

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \nabla_k \quad (2.5)$$

yang dinamakan algoritma penurunan tercuram (*steepest descent*). Namun dalam praktek, gradien  $\nabla_k$  hanya diperoleh secara perkiraan pendekatan. Dengan gradien yang didapat merupakan pendekatan, maka jejak adaptasi menuju titik terendah akan cenderung zig-zag.

Karena perkiraan nilai gradien juga sebenarnya memerlukan pengetahuan mengenai permukaan unjukkerja, maka logis bila diajukan juga algoritma yang hanya memerlukan data atau pengetahuan nilai-nilai sesaat isyarat yang ada, yaitu  $x_k$  dan  $d_k$  atau  $x_k$  dan  $\varepsilon_k = d_k - y_k$ , karena  $y_k$  juga sudah dihitung setelah  $x_k$  masuk sistem adaptif.

Dengan dasar penjabaran algoritma berdasar turunan/diferensial parsial nilai sesaat  $\varepsilon_k^2$ , dan dengan pantauan nilai rerata  $\varepsilon_k^2$  yang telah mencapai nilai kecil atau nilai ambang tertentu, maka muncul algoritma LMS

$$\mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu \mathbf{X}_k \varepsilon_k \quad (2.6)$$

Selain Algoritma LMS ada juga algoritma lain yang dapat digunakan, yaitu Algoritma RLS. Algoritma RLS memiliki perbedaan dengan Algoritma LMS dalam penghitungan bobotnya. Perbedaan antara Algoritma LMS dan RLS dapat dilihat dari rumus bobotnya:  $\mathbf{W}_{j+1} = \mathbf{W}_j + \mathbf{K}_j \mathbf{X}_j \varepsilon_j$ , dengan nilai  $\mathbf{K}_j$  untuk masing-masing Algoritma memiliki nilai yang berbeda, seperti dalam Tabel 1:

**Tabel 1: Nilai  $\mathbf{K}_j$  untuk Algoritma LMS dan RLS**

LMS	$\mathbf{K}_j = 2\mu, \mu_{\max} > \mu > 0$
RLS	$K_j = \frac{P_j}{1 + \mathbf{X}_j^T P_j \mathbf{X}_j}$ $P_{j+1} = P_j - \frac{P_j \mathbf{X}_j \mathbf{X}_j^T P_j}{1 + \mathbf{X}_j^T P_j \mathbf{X}_j}$ $P_0 = \alpha I, \alpha \approx 10^4$

### 3. Metodologi

Untuk pengambilan sinyal, dipasang dua mikrofon untuk memperoleh sinyal referensi di sumber bising dan sinyal utama di dalam ruang yang juga mengandung sinyal bising. Sebagai alat utama adalah komputer dengan spesifikasi: Prosesor Intel Pentium 4, 1 GHz, RAM minimum 128 M, dan *Hard Disk Seagate* minimum 20 GB. *Sound card* yang berfungsi sebagai Analog / Digital (A/D) dan Digital / Analog (D/A) adalah *Creative Vibra* dengan port PCI atau *On Board*. Data yang digunakan adalah bising yang bersifat multifrekuensi dan satu gelombang dengan frekuensi tertentu sebagai data suara yang tidak boleh ikut teredam.

### 4. Hasil dan Pembahasan

Dengan menggunakan komputer yang telah disebutkan spesifikasinya, suara kendaraan direkam dengan menggunakan “*stereo sound card*” melalui dua mikrofon, yang satu diletakkan di dalam ruangan yang akan dihapuskan bisingnya, yang lainnya diletakkan di luar ruangan yang dekat dengan sumber bising, yang bersifat stasioner.

Hasil rekaman suara dari masing-masing mikrofon yang berupa *wave file*,

secara terpisah, selanjutnya diproses dengan algoritma adaptif LMS dan algoritma adaptif RLS. Idealnya galat yang muncul sebagai hasil proses adaptif sama dengan tutur (suara yang tidak boleh ikut teredam). Akan tetapi pada kenyataannya keadaan itu tidak mungkin tercapai. Diteliti seberapa besar galat yang dihasilkan apabila diproses dengan algoritma adaptif LMS dan algoritma adaptif RLS.

#### 4.1 Penghapusan Bising LMS Adaptif Dua Proses Untuk Isyarat Suara Kendaraan

Diteliti nilai MSE untuk suara kendaraan kijang diesel, dengan menggunakan program dalam C, yaitu `latif_3_fx.c`. Mula-mula untuk satu proses adaptif, selanjutnya untuk dua proses adaptif. Hal ini mengingat dengan satu proses adaptif saja masih menyisakan bising yang cukup mengganggu. Penelitian untuk mencari nilai MSE dilakukan dengan memfilter isyarat-isyarat suara kendaraan terlebih dahulu untuk menghilangkan isyarat runcing atau isyarat frekuensi tinggi agar proses adaptif berjalan lebih cepat. Secara rinci nilai-nilai MSE dapat diperhatikan dalam Tabel 2 dan Tabel 3 berikut.

**Tabel 2: Hasil MSE Proses Pertama Suara Kendaraan Kijang Diesel Terfilter Low Pass Filter 15 kHz Dengan Sampel 40.000.**

L=100	d=230	$\mu = 0,001$	MSE1=10,2357
L=230	d=110	$\mu = 0,001$	MSE1=0,1131
L=230	d=100	$\mu = 0,001$	MSE1=0,1200
L=230	d=120	$\mu = 0,001$	MSE1=0,1041
L=230	d=130	$\mu = 0,001$	MSE1=0,1041
L=230	d=140	$\mu = 0,001$	MSE1=0,1049

Tabel 2, menunjukkan bahwa untuk  $\mu=0,001$  dan dengan satu proses

adaptif, nilai MSE minimum terjadi saat  $L = 230$  dengan *delay*  $d = 120$  atau 130. Selanjutnya apabila  $y$  atau *output* proses adaptif pertama menjadi *referens* untuk proses adaptif yang kedua, maka nilai MSE nya dapat dilihat dalam Tabel 3.

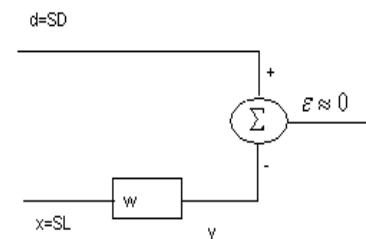
**Tabel 3: Hasil MSE Proses Kedua Suara Kendaraan Kijang Diesel Terfilter Low Pass Filter 15 kHz Dengan Sampel 80.000.**

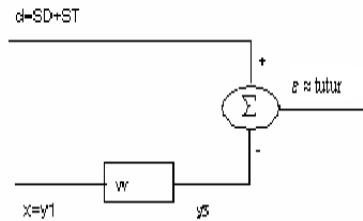
L=100	d=230	$\mu = 0,001$	MSE1=0,2400	MSE2=16,2944
L=230	d=100	$\mu = 0,001$	MSE1=0,0598	MSE2=3,9117
L=230	d=110	$\mu = 0,001$	MSE1=0,0590	MSE2=11,6033
L=230	d=120	$\mu = 0,001$	MSE1=0,0595	MSE2=7,0464
L=230	d=130	$\mu = 0,001$	MSE1=0,1008	MSE2=32,7102

Ternyata nilai optimum dicapai untuk Penghapusan Bising LMS Adaptif dengan dua proses adalah  $L = 230$ , dengan *delay*  $d = 100$ . Nilai inilah yang akan menjadi acuan untuk program LMS Adaptif dengan menggunakan program dalam bahasa C dengan dua proses. Berikut adalah skema program LMS Adaptif dengan menggunakan dua proses:

1. Proses pertama:  $d$  adalah Sinyal Dalam,  $x$  atau referensnya adalah Sinyal Luar, diharapkan *output*nya hampir mirip dengan Sinyal Dalam, sehingga nilai  $\varepsilon$  (*error*) mendekati nol.
2. Proses kedua:  $d$  adalah Sinyal Dalam dicampur tutur, sedang  $x$  diambil *output* dari proses pertama. Diharapkan nilai  $\varepsilon$  (*error*) lebih mendekati nol.

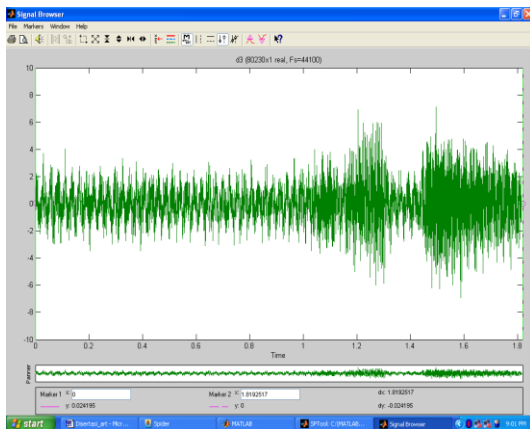
Untuk lebih jelasnya diberikan skema penelitian dengan dua proses diatas:



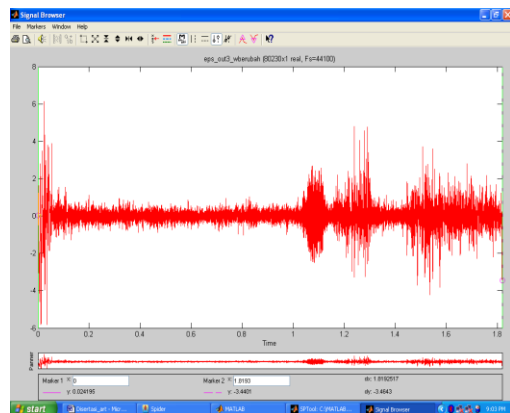


Gambar 3.1: Bagan Penelitian LMS Adaptif dengan Dua Proses

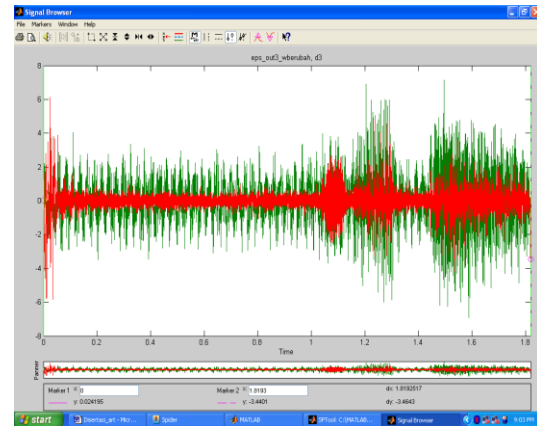
Hasil kedua proses diatas dapat dilihat pada Gambar 3.2.a sampai dengan 3.2.c, dengan program C yaitu Sidang2\_2\adaptif.c. Hasil program tersebut disimpan dalam sptool (Matlab), yaitu dalam C\work\Disertasi\_2\Frek Sampel Berubah\ Sidang2\_2.spt.



Gambar 3.2.a: Isyarat Tutur Bercampur dengan Bising Kendaraan Diesel Sebelum Dilakukan Kedua Proses LMS Adaptif

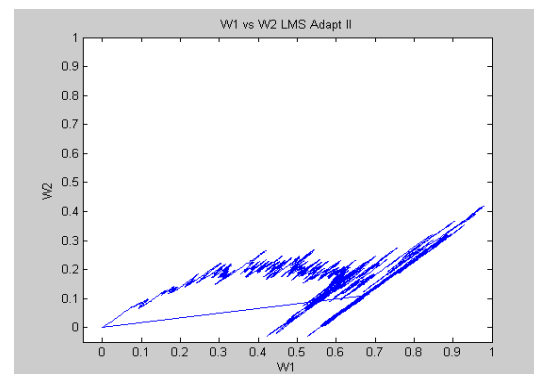


Gambar 3.2.b: Isyarat Tutur Bercampur dengan Bising Kendaraan Diesel Setelah Dilakukan Kedua Proses LMS Adaptif



Gambar 3.2.c: Isyarat Tutur Bercampur dengan Bising Kendaraan Diesel Sebelum dan Setelah Dilakukan Kedua Proses LMS Adaptif

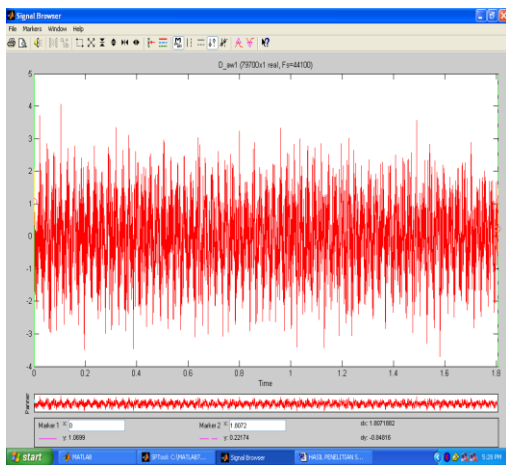
Gambar 3.2.c menunjukkan sudah terdapat penghapusan bising dari isyarat asli tutur yang bercampur bising (isyarat berwarna hijau) dan hasil proses LMS Adaptif dengan dua proses (isyarat berwarna merah) dengan  $d = 100$  dan  $L = 230$ . Apabila didengarkan dengan telinga manusia, tutur tetap terdengar. Meskipun tutur yang berfrekuensi rendah ikut teredam, namun tutur yang berfrekuensi tinggi masih terdengar jelas. Sisa bising masih terdengar tetapi sudah bukan merupakan suara kendaraan tetapi suara desis. Adapun perjalanan bobot  $w_1$  dan  $w_2$  pada proses II (program matlab: ralat \_delay \_awal \_ iterasi \_0, Lampiran 8) terdapat dalam Gambar 3.2.d.



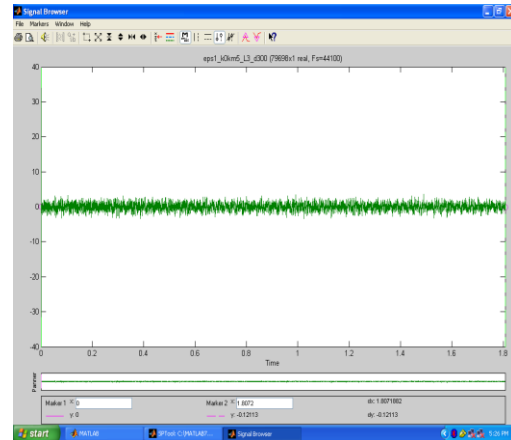
Gambar 3.2.d: Koordinat  $W_1$  dan  $W_2$  Hasil Proses LMS Adaptif Kedua

## 4.2 Penghapusan Bising RLS Adaptif Dua Proses Untuk Isyarat Suara Kendaraan

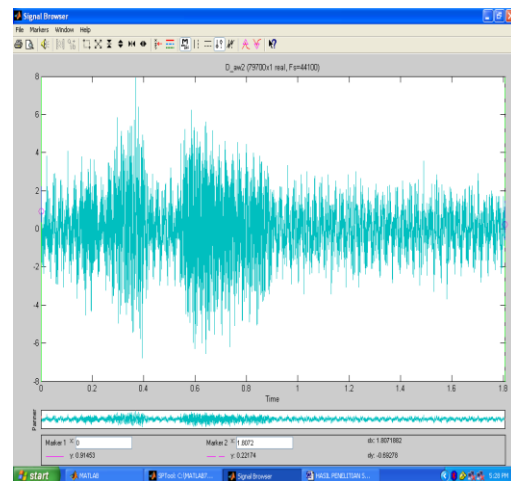
Proses RLS Adaptif cukup berat komputasinya untuk  $L$  yang besar, hal ini disebabkan karena dalam algoritmanya terdapat matriks  $L \times L$  yang dihitung sampai iterasi ke  $N$ . Setelah dilakukan proses RLS Adaptif prediktif, diharapkan dengan memperbesar  $L$ , maka nilai konstanta dalam penyebut rumus  $\mathbf{K}$  (atau selanjutnya dinamakan  $\mathbf{var\_K}$ ) akan dapat mengecil. Akan tetapi kenyataannya makin besar  $L$  hasil adaptasinya tidak menunjukkan adanya keberhasilan dalam peredaman bising, akan tetapi makin meningkatkan bising baru yang memiliki frekuensi tinggi, sehingga kedengaran seperti bunyi desis. Hal ini cukup mengganggu pendengaran manusia. Karena kondisi itulah, dilakukan dua proses adaptif dengan  $L$  cukup kecil, yaitu  $L = 3$ .



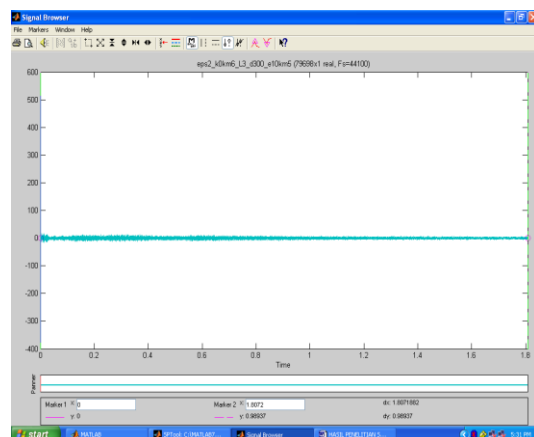
Gambar 3.3.a: d Pada Proses RLS Adaptif I



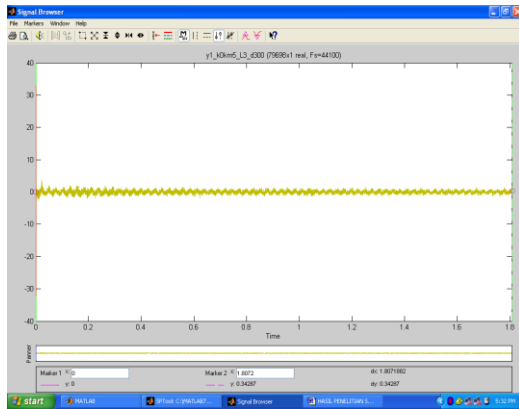
Gambar 3.3.b: Hasil  $\mathcal{E}$  Pada Proses I dengan  $L = 3$ ,  $\mathbf{var\_K}=0,5$ , Delay 300



Gambar 3.3.c: d Pada Proses RLS Adaptif II (dengan Tutur)

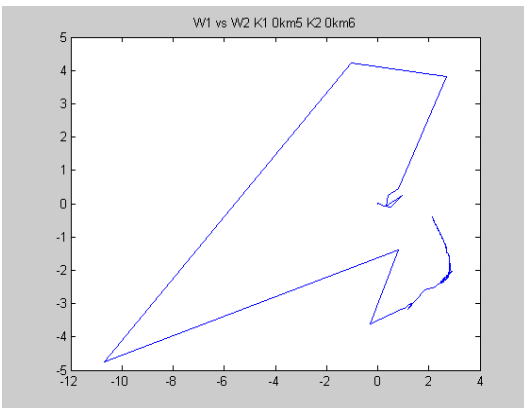


Gambar 3.3.d: Hasil  $\mathcal{E}$  Pada Proses II dengan  $L = 3$ ,  $\mathbf{var\_K}=0,6$ ; Delay 300



Gambar 3.3.e: Hasil  $y$  Pada Proses I dengan  $L = 3$ ,  $\text{var}_K=0,5$ , Delay 300

*Output*  $y$  mirip dengan  $d$ , meskipun masih nampak ada *delay*. Gambar yang lebih jelas terdapat pada Sptool pada folder RLS\_lengkap, dengan nama RLS\_2proses. Maka *output*  $y$  dapat dijadikan referensi untuk proses Adaptif RLS II, dengan  $d$  adalah isyarat dari dalam ruang yang telah dicampur tutur. Selanjutnya akan diberikan perjalanan bobot  $w_1$  dan  $w_2$  untuk proses Adaptif RLS pada proses II.



Gambar 3.3.f: Perjalanan Bobot  $W_1$  dan  $W_2$  Pada Proses RLS Adaptif II

Dari uraian di atas dapat disimpulkan dalam Tabel 4, hasil perbedaan antara LMS Adaptif dengan RLS Adaptif.

**Tabel 4: Hasil Penghapusan Bising dengan Berbagai Macam Proses Adaptif**

Algoritma	Delay	Nilai Optimal	L Optimal	Perjalanan Bobot $W_1, W_2$ , Proses terakhir
LMS	100	$\mu = 0,001$	$L=230$	Koordinat X: 0 sd 1 Y: -0,1 sd 0,4
RLS	300	Proses I: $\text{var}_K=0,5$ Proses II: $\text{var}_K=0,6$	$L=3$	Koordinat X: -12 sd 4 Y: -5 sd 5

Pada Tabel 4, *delay* yang terbanyak terdapat pada proses RLS, yaitu 300, hal ini disebabkan karena  $L$  pada proses RLS bernilai kecil.  $L$  yang besar pada proses RLS menyebabkan waktu komputasi yang sangat panjang, karena adanya perhitungan matriks  $L \times L$ . Hasil adaptasinya untuk  $L$  besar pun akan menjadi tidak optimal karena hilangnya bunyi bising digantikan oleh munculnya desis yang cukup besar, yang diakibatkan karena adanya *error* komputasi. Dalam Algoritma RLS ini penghitungan *delay* cukup membantu mengoptimalkan hasil adaptasi, karena  $L$  dalam Algoritma ini tidak memungkinkan diambil besar seperti Algoritma LMS.

#### 4.3 Perbandingan Antara Algoritma LMS Adaptif dan RLS Adaptif

Hasil perjalanan pasangan bobot  $W_1$  dan  $W_2$  juga menempati luasan yang berbeda. Algoritma LMS memerlukan luasan sekitar 0,5 satuan luas untuk perjalanan pasangan bobotnya. Algoritma RLS sebesar 160 satuan luas. Meskipun perjalanan bobot untuk Algoritma RLS cukup luas, akan tetapi sebenarnya Algoritma ini hanya memerlukan beberapa langkah perjalanan bobot untuk mencapai nilai optimum (dilihat dari nilai  $L$  yang cukup kecil), meskipun perjalanan bobot itu

memiliki jarak tempuh yang cukup panjang. Hasil  $\varepsilon$  untuk Algoritma LMS (Gambar 3.2.c) telah menunjukkan adanya penurunan suara bising sebelum dan setelah diredam. Selain itu suara informasi yang diinginkan tetap muncul meskipun ada pengurangan amplitude untuk informasi dengan frekuensi rendah. Hal ini bahkan dapat menjadi keuntungan (nilai positif) karena informasi dengan frekuensi yang lebih tinggi akan membuat telinga manusia lebih jelas dalam menerima informasi tersebut.

## 5. Kesimpulan

Dari penelitian di atas dapat disimpulkan bahwa

1. Algoritma LMS Adaptif memerlukan nilai  $L$  yang lebih besar daripada Algoritma RLS Adaptif.
2. Meskipun lebih lama namun Algoritma LMS Adaptif dapat meredam bising lebih baik daripada Algoritma RLS Adaptif.
3. Perjalanan bobot Algoritma LMS menempati luasan yang jauh lebih kecil dibandingkan dengan Algoritma RLS.

## Daftar Pustaka

- Sumeh, Alat Peredam Bising Elektronis, Tesis, S2 Elektro UGM, 1999.
- Widrow, B., dan S.D. Stearns, *Adaptive Signal Processing*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1985.
- Prastiyanto, D., Adhi S, A., Thomas S W, T., Simulasi Identifikasi Kanal Lintas-Jarak pada WCDMA secara Adaptif dengan Filter NLMS dan LMS, *Telkomnika*, Vol. 4, No. 1, April 2008.
- Nascimento, V.H., *The Normalized LMS Algorithm with Dependent Noise*, Departamento de Engenharia de Sistemas Electronicos, Escola Politecnica, Universidade de Sao paulo, Sao Paulo, SP.
- Susanto, A., Research Report, 1982., *Comparison of Three Main Adaptive Algorithm.*, UC Davis., University of California.